Available online at www.sciencedirect.com
SCIENCE DIRECT
Journal of Symbolic Computation 41 (2006) 30–48
Journal of Symbolic Computation
www.elsevier.com/locate/jsc

Local parametrization of cubic surfaces

Ibolya Szilágyi[a,*], Bert Jüttler[b], Josef Schicho[c]

[a] *RISC–Linz/RICAM, Johannes Kepler University, A-4040 Linz, Austria*
[b] *Institute of Applied Geometry, Johannes Kepler University, A-4040 Linz, Austria*
[c] *RICAM, Austrian Academy of Sciences, A-4040 Linz, Austria*

Received 29 June 2004; accepted 30 June 2005
Available online 30 September 2005

Abstract

Algebraic surfaces – which are frequently used in geometric modelling – are represented either in implicit or parametric form. Several techniques for parametrizing a rational algebraic surface as a whole exist. However, in many applications, it suffices to parametrize a small portion of the surface. This motivates the analysis of local parametrizations, i.e., parametrizations of a small neighborhood of a given point P of the surface S. In this paper we introduce several techniques for generating such parametrizations for nonsingular cubic surfaces. For this class of surfaces, it is shown that the local parametrization problem can be solved for all points, and any such surface can be covered completely.

Keywords: Parametrization; Cubics; Algorithm; Surfaces

1. Introduction

Many techniques from geometric modelling and Computer Aided Design are based on algebraic surfaces. Typically, these surfaces are described as the zero set of an algebraic equation (implicit representation), or as the image of map given by rational functions (parametric representation). Since both representations are appropriate for solving different types of problems, the automatic transition between these two representations is very important.

For instance, surface/surface-intersections can be traced efficiently if one of the surfaces is given in implicit form, and the other in parametric form. Another example is the detection of

* Corresponding author. Tel.: +43 732 2468 9950; fax: +43 732 2468 9930.
E-mail addresses: szibolya@risc.uni-linz.ac.at (I. Szilágyi), Bert.Juettler@jku.at (B. Jüttler), josef.schicho@oeaw.ac.at (J. Schicho).

0747-7171/$ - see front matter
doi:10.1016/j.jsc.2005.06.001

self-intersections of a surface, which becomes much simpler if both representations (implicit and parametric) are available. Algebraic methods for enhancing the performance of intersection algorithms in Computer-Aided Design are currently under investigation in a European project (Dokken et al., 2001).

Various techniques for generating a rational parametric representation of rational algebraic surfaces (called *parametrization* for short) are available, see Bajaj et al. (1998), Schicho (1998b), Sederberg and Snively (1987) and Pérez-Díaz et al. (2005). The reverse process is called *implicitization*. The implicitization problem is always solvable, and there are several different approaches to deal with this problem, as described by Busé (2001), Buchberger (1988), Corless et al. (2001), Dokken (2001) and Zheng et al. (2003).

This paper is devoted to general cubic surfaces, which have both an implicit and a rational parametric representation, except for the cone over an elliptic planar cubic curve. This property may make them particularly useful in a number of geometric modelling operations. On the other hand, these surfaces are sufficiently general, since any real-valued function on $\mathbb{R}^3$ can efficiently be approximated by a piecewise cubic function which is continuously differentiable, using three-dimensional Clough–Tocher elements, see Hoschek and Lasser (1993).

In most cases, the existing parametrization methods produce a birational map. Many methods use the 27 lines on a nonsingular cubic surface for parametrizing it (Berry and Patterson, 2001; Sederberg and Snively, 1987). It should be noted, however, that the computation of the lines is not a simple problem (Bajaj et al., 1998; Sederberg, 1990).

Several parametrization methods cannot be applied to surfaces with two real components. In such situations, one either uses two disjoint parametrizations or a two-to-one parametrization (Sederberg and Snively, 1987). Since the mentioned parametrization methods can be used only for certain classes of cubics, a thorough case analysis is needed.

Algebraic techniques often parametrize the algebraic surface as a whole. In many applications (such as geometric modelling and related areas), however, it suffices to have a parametrization defined in some open subset in the parameter space that covers the intersection of the surface with a certain region of interest. In contrast to the classical problem, we will refer to this as the problem of *local parametrization*: find a parametrization of a small neighborhood of a given point of the surface.

In this paper, we give a method for computing local parametrizations of non-singular cubic surfaces. The method works without analyzing the system of lines on the cubic surface. It produces rational maps defined in some neighborhood of the origin in the plane with the property that the image is an open subset of a given nonsingular cubic surface containing a given point P.

If the coefficients of the given surface are rational numbers then the coefficients of the computed local parametrization are real algebraic numbers. These have to be represented somehow, for instance using Thom's code (Coste and Roy, 1988). In CAGD applications one prefers to work with floating point numbers, taking into account the existence of small numerical errors. In particular the reference point P on the cubic surface S, which is part of the input, is probably not known exactly as a real algebraic number, but approximately as a finite floating point expansion. The output is also required to be in term of floating point numbers.

This imposes a question: *Is there a justification to apply an exact algorithm to floating point data?* In general, there is not, and such applications may lead to all kinds of failures (see Schirra (2000) for an extended discussion). Methodologically, we translate a mathematical/exact solution to an engineering/approximate solution, and something can go wrong in the process of the translation because the mathematical model is never absolutely accurate. Often enough, the translation is sufficiently accurate to produce useful results, and we will give some evidence that this is particularly the case for our algorithm. But the question for a mathematical justification remains.

In general an algorithm is considered to be robust if it produces the correct result for some perturbation of the input. Hence, one way to prove the robustness of an algorithm is to show that there is always an input for which the algorithm gives the correct results. However, this is very often a complicated task. A complete error analysis of an algorithm even in more simple problems can be cumbersome, see Pérez-Díaz et al. (2004). As the presented algorithms are much too complicated, a detailed error analysis is beyond of the scope of the present paper.

A weaker justification would be to observe, that all operations and manipulations on the level of coefficients in our algorithms are continuous. This means that small numerical errors in the input and in the computations lead to small numerical errors in the output. The resulting local parametrization and the intermediate results are not exact, but they are close to an exact result with real algebraic coefficients. In the case of a continuous algorithm if we increase the precision, the computed result converges to the exact result. However, we have to admit that we failed to make all operations continuous in our algorithms. The difficulties are explained in more detail in Section 4.6.

Our justification that the provided algorithms work on floating point data is empirical. Experiences show that increasing the precision in the computation leads to more and more accurate results. For further details, see Sections 4.2 and 4.4.

We use three local parametrization techniques for cubic surfaces, which are called the two-curve technique, the repeater technique, and the reflection technique. The first two techniques can be traced back to Manin (1986) and Abhyanker and Bajaj (1987). They are based on the classical theory of rational curves on cubic surfaces. Such curves may be generated as the intersection of the surface with the tangent plane at a generic surface point.

We give a complete geometrical analysis of the introduced techniques for nonsingular cubic surfaces, and we show that each of the three algorithms computes a local parametrization for a given nonsingular cubic surface S, and a surface point P. The computed parametrization is improper. Clearly, properness cannot be expected, since the so-called F_5 surface has no proper parametrization (Schicho, 1998a). No computation of the lines on the surface is needed.

The remainder of the paper is organized as follows. Section 2 recalls some basic facts about cubic surfaces and introduces the local parametrization problem. Section 3 is devoted to a certain property of surface points, which we call the "t-property". The three algorithms for local parametrization are described in Section 4. We analyze each technique and we show that each provides a local parametrization around a given surface point. Finally, Section 5 concludes the paper.

2. Preliminaries

After recalling some properties of cubic surfaces, we introduce the notion of local parametrizations.

2.1. Cubic surfaces

Throughout this paper we work in the real projective space. We will consider a nonsingular cubic surface S. It is given by its implicit form F. A point of the surface will be called *generic*, if it does not belong to one of the lines lying completely on the surface.

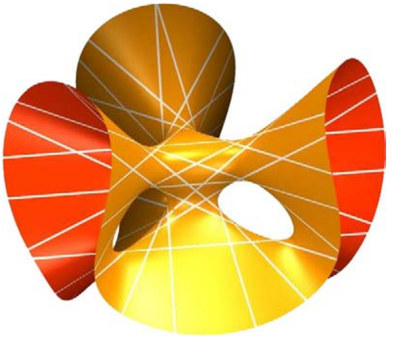

Fig. 1. F_1 surface.

Cubic surfaces are the zero set of a polynomial of degree three. It has been known since 1849, when Cayley and Salmon published their famous theorem, that there are 27 lines lying completely on a nonsingular cubic surface. One may conclude this theorem from the fact that the number of lines on a nonsingular cubic surface is equal to the number of double tangent planes of an arbitrary tangent cone to the surface (Henderson, 1960).

Schläfli classified the cubic surfaces with respect to the number of real lines on them. The non-singular cubic surfaces can be divided into five types $F_1, F_2, \ldots, F_5$ with respect to the number of real lines (27, 15, 7, 3 and 3, respectively) and real components (1, 1, 1, 1 and 2, respectively).

Later, Schläfli classified the cubic surfaces (singular and non-singular ones) into 23 species with respect to the nature of the singularities on the surfaces. A complete classification with 21 classes over $\mathbb{C}$ has been given by Bruce and Wall (1979).

For future reference we recall that each non-singular cubic surface has at least one real line, and that surfaces consist of one ($F_1, \ldots, F_4$) or two (F_5) real components. One of the two components of the F_5 surface is convex in the following sense:

Definition 1. A connected component of a surface is said to be convex, if there exists an auxiliary plane, such that for any tangent plane of the component, the component is fully contained in one of the two cells defined by the planes.

The auxiliary plane acts as the plane at infinity.

Fig. 1 shows a surface with one real component and all the real lines on it, and Fig. 2 shows a cubic surface of type F_5 with two components (both pictures courtesy of O. Labs).

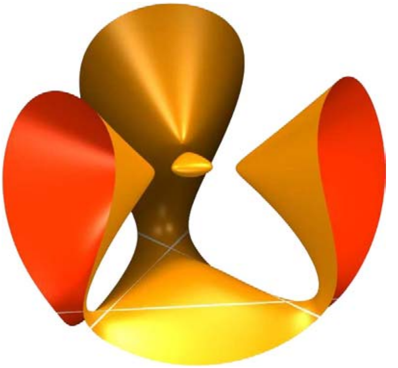

Fig. 2. F_5 surface.

2.2. Local parametrization

Given the surface S and a point $P = (p_1 : \cdots : p_4)$ on it, we are interested in finding a rational map defined in a certain neighborhood of the origin, which is "well-behaved" at P, and covers a certain neighborhood of the given point.

Definition 2. A quadruple of polynomials $(\pi_1(u, v), \ldots, \pi_4(u, v))$ is called a local parametrization of the surface S at the point P, if the image of the origin is P,

$$(\pi_1(0, 0) : \pi_2(0, 0) : \pi_3(0, 0) : \pi_4(0, 0)) = (p_1 : p_2 : p_3 : p_4), \tag{1}$$

and the image of the rational map defined by the four polynomials is fully contained in the surface. The local parametrization is said to be regular, if the Jacobian matrix of the mapping

$$(u, v, \rho) \mapsto (\rho\, \pi_1(u, v), \ldots, \rho\, \pi_4(u, v)) \tag{2}$$

has full rank (i.e., 3) at $(0 : 0 : 1)$.

The following result is an immediate consequence of the implicit mapping theorem, see e.g. Kendig (1977).

Proposition 3. *For any given regular local parametrization G there exists a neighborhood of the origin in the parameter space, such that the restriction of G to this neighborhood is faithful.*

3. Analyzing the system of all tangent planes

We analyze the location of points with respect to the system of all tangent planes of the given cubic surface.

3.1. The t-property

We introduce the following auxiliary notion.

Definition 4. Let S be a cubic surface and $P \in S$ a generic point on the surface. P is said to have the t-property if P is contained in the tangent plane at another surface point.

Clearly, this second tangent plane is different from the tangent plane at P.

We recall the definition of *contour generator* and *apparent contour* following Cipolla and Giblin (2000).

Using a perspective projection, we project a given surface $\bar{S}$ from a given point P into a plane Π, $P \notin \Pi$. The point P is called the *center of the projection*, while Π acts as the *image plane*.

We consider the cone of lines through P which are tangent to the surface $\bar{S}$. This cone is called the *tangent cone* to $\bar{S}$ with apex P. The curve on $\bar{S}$ where this cone is tangent to $\bar{S}$ is called the *contour generator*, and the curve where the cone intersects the image plane is the *apparent contour*.

In the case of a cubic surface S, the contour generator is a space curve of degree 6, and the apparent contour is a planar quartic curve. In fact, if we move the point P (i.e., the center of the projection) to the origin $(0 : 0 : 0 : 1)$, the equation of the surface takes the form

$$x_4^2 L(x_1, x_2, x_3) + x_4 Q(x_1, x_2, x_3) + K(x_1, x_2, x_3) = 0, \tag{3}$$

where L, Q and K are linear, quadratic and cubic homogeneous polynomials, respectively. After a short computation one arrives at the equation

$$[Q(x_1, x_2, x_3)]^2 - 4L(x_1, x_2, x_3)\, K(x_1, x_2, x_3) = 0 \tag{4}$$

of the apparent contour $\mathcal{C}_P$. The linear form is L is the equation of the line arising from the intersection of the tangent plane $T_P S$ with the image plane Π.

First we analyze the singularities which may be present in the apparent contour.

Lemma 5. *The apparent contour associated with a point P on a non-singular cubic surface has a singular point if and only if the point P lies on one of the lines on the surface.*

Proof. Let $P = (0 : 0 : 0 : 1)$, and assume that the equation of the surface has the form (3). We may assume that $L = x_3$, i.e., that the tangent plane at P is $x_3 = 0$.

First case: The apparent contour has a singular point in the tangent plane at P. Without loss of generality we assume that it is located at $(1 : 0 : 0)$. A short computation reveals that this implies $q_{2,0,0} = k_{3,0,0} = 0$, where q_{ijk} and k_{ijk} are the coefficients of $x_1^i x_2^j x_3^k$ in Q and K, respectively. Consequently, the line $(s : 0 : 0 : t)$ $(s, t \in \mathbb{R})$ is fully contained in the surface.

Second case: The apparent contour has a singular point which is not in the tangent plane at P. Without loss of generality we assume that it is located at $(0 : 0 : 1)$. A short computation reveals that this implies $k_{0,1,2} = \frac{1}{2} q_{0,0,2} q_{0,1,1}$, $k_{1,0,2} = \frac{1}{2} q_{0,0,2} q_{1,0,1}$ and $k_{0,0,3} = \frac{1}{2} q_{0,0,2}^2$. The surface is singular, since it has the singular point $(0 : 0 : -2 : q_{0,0,2})$.

Finally, it can be shown that any line through P generates a singular point of the apparent contour. □

The t-property can now be characterized by using the apparent contour.

Proposition 6. *A generic point P of the cubic surface S has the t-property if and only if the apparent contour of the surface with center P has real points.*

Proof. The point P has the t-property, if and only if there exists a point $R \in S$, $R \neq P$, such that $P \in T_R S$, where $T_R S$ is the tangent plane to the surface S at R. This is equivalent to the fact that the line connecting P and R is a real line of the tangent cone with apex P. This line corresponds to a regular point of the apparent contour. Note that the apparent contour cannot have singularities, since P is assumed to be a generic point (cf. Lemma 5). □

The following algorithm, which is based on Proposition 6, is needed for computing the local parametrizations, as described in the next sections.

Algorithm 1 (``point on contour'')
Given: An implicit equation F of a cubic surface S and a general point $P \in S$.
Synopsis: Decide the t-property for P. If P has the t-property find $R \in S$, such that $P \in T_R S$.

(1) We move P to the origin by a linear transformation of the homogeneous coordinates such that the tangent plane at P becomes $x_3 = 0$, and compute the equation of the apparent contour (4).
(2) Check whether the apparent contour has real points using methods similar to Gonzalez-Vega and Necula (2002).
  (a) If the apparent contour does not have real points, then the algorithm stops. The point P does not have the t-property.
  (b) If the apparent contour has non-singular real points, then P has the t-property. Go to the next step.
(3) Find a real point $\bar{R}_a = (\bar{x}_1 : \bar{x}_2 : \bar{x}_3)$ on the apparent contour such that $\bar{x}_3 \neq 0$.

(4) Compute the corresponding point R on the contour curve. If P is at $(0:0:0:1)$, then

$$\bar{R} = \left(\bar{x}_1 : \bar{x}_2 : \bar{x}_3 : \frac{-Q(\bar{x}_1, \bar{x}_2, \bar{x}_3)}{2L(\bar{x}_1, \bar{x}_2, \bar{x}_3)}\right). \tag{5}$$

(Note that $L(\bar{x}_1, \bar{x}_2, \bar{x}_3) = \bar{x}_3 \neq 0$.)

(5) Using the inverse of the linear transformation in (1), we transform $\bar{R}$ to get $R \in S$.

Step 1 Assume the point P is not at infinity and the tangent plane at P has an equation $a_1x_1 + \cdots + a_4x_4 = 0$ with $a_3 \neq 0$. Indeed, in this situation we can use the projective transformation given by the matrix

$$\begin{pmatrix} 1 & 0 & 0 & -\frac{p_1}{p_4} \\ 0 & 1 & 0 & -\frac{p_2}{p_4} \\ \frac{a_1}{a_3} & \frac{a_2}{a_3} & 1 & \frac{a_4}{a_3} \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{6}$$

If one of these conditions does not hold, we relabel coordinates. This should also be done if p_4 or a_3 is close to zero (i.e. if its order of magnitude is the same as the precision).

Step 3 Finding a real point on a nonsingular real algebraic curve can be done by computing a real solution between the real zeros of its discriminant.

3.2. Locating the points with the t-property

Given a non-singular cubic surface, we identify the regions of points with and without the t-property on a nonsingular cubic surface.

Lemma 7. *The regions on a cubic surface S containing points with and without the t-property are bounded by the real lines of S.*

Proof. Consider the system of apparent contours associated with all points on the surface. Clearly, the coefficients of these planar curves depend continuously on the location of the points.

If one moves along a curve from a point P with the t-property to a point Q without it, the apparent contour, which has at least one real component at P, has first to degenerate to a singular point, before disappearing eventually. Due to Lemma 5, this takes place exactly when one crosses one of the lines lying on the surface. □

Depending on the local behavior of a surface with respect to its tangent plane at a point, one arrives at different types of surface points. We assume that we have a non-flat surface point, i.e., $Q \not\equiv 0$ in (3). A point is called *elliptic* if the tangent plane at the point intersects the surface (locally) in an isolated point, *hyperbolic* if the tangent plane intersects the surface (locally) in a pair of intersecting curves with two different tangents, and *parabolic* otherwise.

One may distinguish the three types of points by the Gaussian curvature K. A point of a surface is called *elliptic* if $K > 0$, *parabolic* if $K = 0$, and *hyperbolic* if $K < 0$.

The non-convex component of a cubic surface may consist of hyperbolic, elliptic, and parabolic points, while the convex component of the F_5 surface has elliptic points only.

Lemma 8. *Generic hyperbolic points of a cubic surface have the t-property.*

Proof. A short computation reveals that the two asymptotic directions at a generic hyperbolic point (i.e., the tangent directions of the two branches of the intersection curve with the tangent plane at the point) correspond to real points of the apparent contour. □

Theorem 9. *A generic point of a nonsingular cubic surface S has the t-property if and only if it lies on the non-convex component of the surface.*

Proof. Any non-singular cubic surface contains at least three real lines. Lines are always on the non-convex component of S, as the convex component does not contain any line. These lines define a partition of the component into several cells.

Any line of a nonsingular cubic surface contains only hyperbolic points, with the exception of the two parabolic points (Segre, 1942). Consequently, the neighborhood of any line contains hyperbolic points which have the t-property (Lemma 8). According to Lemma 7, if a point has the t-property, then this property is shared by all points in the cell.

It remains to be shown that the convex component of the F_5 surface does not contain points with the t-property. Consider any point $R \in S$ on the convex component. Assume that there is a point $Q \in S$ such that $R \in T_Q S$. As the tangent plane cannot intersect the convex component in a different point than Q, the point Q cannot lie on the convex component. The line in $T_Q S$ connecting Q and R has four intersections with the surface S, since Q has to be counted twice. This is a contradiction, since any line has at most three real intersections with a cubic surface. □

4. Three techniques for generating local parametrization

We describe three approaches to the solution of the local parametrization problem. The three techniques are based on the theory of rational curves on cubic surfaces. For the convenience of the reader, we summarize it in the next section.

4.1. Rational cubics on cubic surfaces

The intersection of a cubic surface with the tangent plane at a generic surface point P always gives a rational planar cubic, where the point will be the singular point of the curve. A rational cubic can be parametrized by a pencil of lines through the singularity of the curve, which intersect the cubic at exactly one other point. The coordinates of the latter point give parametric functions for the cubic curve.

More precisely, if we assume that $P = (0 : 0 : 0 : 1)$ and that the tangent plane at the origin equals $x_3 = 0$, the equation of the surface takes the form

$$x_4^2 x_3 + x_4 Q(x_1, x_2, x_3) + K(x_1, x_2, x_3) = 0.$$

The cubic curve C_P cut by the tangent plane at the origin is

$$Q(x_1, x_2, 0)x_4 + K(x_1, x_2, 0) = 0.$$

It has the rational parametrization $(Q(1, t, 0) : t \cdot Q(1, t, 0) : 0 : -K(1, t, 0))$. See Abhyankar and Bajaj (1988) for further details.

4.2. The two-curve technique

This technique has been described by Manin (1986). Let Q_1 and Q_2 be two real points on the cubic surface S as in Fig. 3. We denote by C_{Q_i} the curves cut by the tangent plane $T_{Q_i} S$,

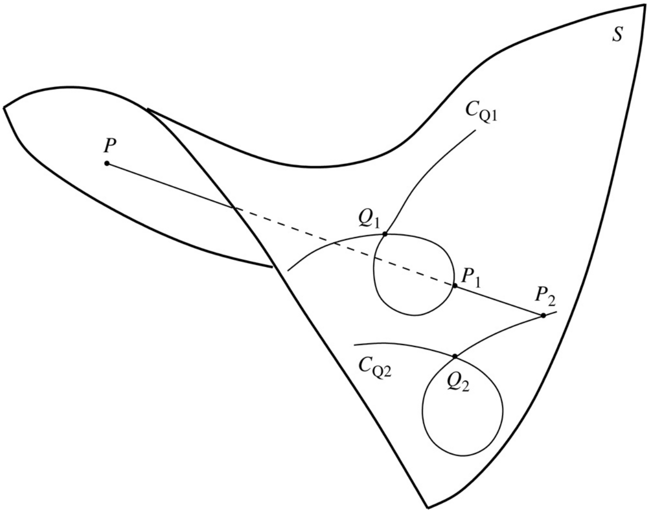

Fig. 3. The two-curve technique.

$i = 1, 2$, from the surface S. The cubic curves C_{Q_i} have a double point at Q_i, therefore they can be parametrized by rational functions.

Let $\pi_i : \mathbb{R} \to C_{Q_i}$ be the parametrization of the ith curve. Then $\pi : \mathbb{R}^2 \to S$, $(t_1, t_2) \mapsto P$ gives a parametrization of a neighborhood of P, where P is the third point of the surface obtained by intersecting with the line $\pi_1(t_1), \pi_2(t_2)$.

This idea is formalized in the following algorithm.

Algorithm 2 (``two-curve technique'')
Given: An implicit equation F of a cubic surface S and a point $P \in S$.
Synopsis: Find four polynomials depending on two parameters, which define a local parametrization of S around P.

(1) Check the t-property for P
   (a) If P does not have the t-property the algorithm stops.
   (b) If P has the t-property go to the next step.
(2) Choose a line through P which has two further intersections P_1, P_2 with S enjoying the t-property.
(3) Compute the intersection points P_1, P_2.
(4) (a) Choose a point Q_1 on the contour of P_1 using Algorithm 1.
   (b) Choose a point Q_2 on the contour of P_2 such that the tangents at P_1 and P_2 to the curves C_{Q_1} and C_{Q_2} are not coplanar.
(5) Parametrize the cubics $C_{Q_i} = S \cap T_{Q_i} S$, such that the parameter 0 corresponds to P_i, $i = 1, 2$.
(6) Let the parametrization of C_{Q_i} be $(x_i(t_i) : y_i(t_i) : z_i(t_i) : w_i(t_i))$. Intersect the line $(x_1(t_1) + \lambda x_2(t_2) : y_1(t_1) + \lambda y_2(t_2) : z_1(t_1) + \lambda z_2(t_2) : w_1(t_1) + \lambda w_2(t_2))$ with S; this leads to a quadratic equation with one root at 0. Compute the remaining root $\lambda(t_1, t_2)$ and substitute it back into the equation of the line. This gives the parametrization of the surface S around the point P.

We give a more detailed description for some steps of Algorithm 2.

Step 3 Let the line l_r^P through P is given by $(p_1 + \mu \cdot r_1 : \cdots : p_4 + \mu \cdot r_4)$. To compute the intersection of S and l_r^P we substitute the equation of the line into the equation of the surface $F(l_r^P)$, and compute the solutions for μ. Since one of the solutions is equal to zero, we divide by μ and obtain a quadratic equation. Solving it we get the further intersection points of S and l_r^P.

Step 4 Intersecting the tangent planes T_{Q_1} and T_{P_1} gives the tangent l_{P_1} at P_1 to the curve C_{Q_1}. To get the forbidden tangent line $\bar{l}_{P_2}$ at P_2, we intersect the tangent plane at the point by l_{P_1}.

We determine the tangents from the point $\bar{l}_{P_2} \cap \Pi$ to the curve $\mathcal{C}_{P_2}$. The points on the apparent contour determined by these tangent directions are the points that should be avoided in the process of determining Q_2 in step 4(b) of the algorithm. (The maximum number of such points is 12.)

Step 5 As the last step of the algorithm, we have to intersect the line $C_{1,x_i} + \lambda C_{2,x_i}$ with S, and compute the values of λ. After substituting the equation of the line into F we get $B_1(t_1, t_2)\lambda + B_2(t_1, t_2)\lambda^2$. Hence, $\lambda = -B_1/B_2$.

We apply the algorithm to an example.

Example 1. Consider the surface S defined by

$$F = 3x_4x_1^2 + 3x_4x_2^2 + 3x_4x_3^2 - 10x_1x_2x_3 - 3x_4^3,$$

and a point $P = (1 : 3 : 27 : 27)$ on S. Using Algorithm 1 we check that the point P has the t-property. We take a line through P and intersect it with the surface S, giving two additional points on S:

$$P_1(-5.98687 : -3.36937 : -1.87809 : -5.09084),$$
$$P_2(-1.07744 : -0.44309 : 1.44983 : 0.83569).$$

We want to compute two points Q_1, Q_2, such that P_i is on the tangent plane $T_{Q_i}S$. For this we have to compute a point on the contour curve of S with respect to the projection from P_i.

The apparent contour of S with respect to the projection from P_1 is:

$$\begin{aligned} K_1 = {} & (-15.27251x_1^2 - 15.27251x_2^2 - 15.27251x_3^2 + 18.78088x_1x_2 + 33.69367x_1x_3 \\ & + 59.86867x_2x_3)^2 + 40(119.58934x_1 - 9.52124x_2 - 144.35332x_3)x_1x_2x_3. \end{aligned}$$

$Q_1^p(6.53995 : 5.94157 : -2.97076)$ is a point on K_1, which corresponds to the point

$$Q_1(0.55308 : 2.57216 : -4.84885 : -5.09084)$$

on S. The intersection of S with the tangent plane at Q_1 gives a curve C_1. The parametrization of C_1 is:

$$\begin{pmatrix} 2.15038t_1^3 - 44.54884t_1^2 + 216.56989t_1 - 263.60396 \\ 4.48066t_1^3 - 74.41867t_1^2 + 205.08880t_1 - 148.34822 \\ -18.79020t_1^2 + 75.48726t_1 - 82.69451 \\ -24.43222t_1^2 + 160.93907t_1 - 224.14926 \end{pmatrix}.$$

The apparent contour of S with respect to the projection from P_2 is:

$$\begin{aligned} K_2 = {} & (1.25354x_1^2 + 1.25354x_2^2 + 1.25354x_3^2 - 7.24916x_1x_2 + 2.21548x_1x_3 \\ & + 5.38720x_2x_3)^2 + 40(0.25543x_1 + 3.34983x_2 + 0.62389x_3)x_1x_2x_3. \end{aligned}$$

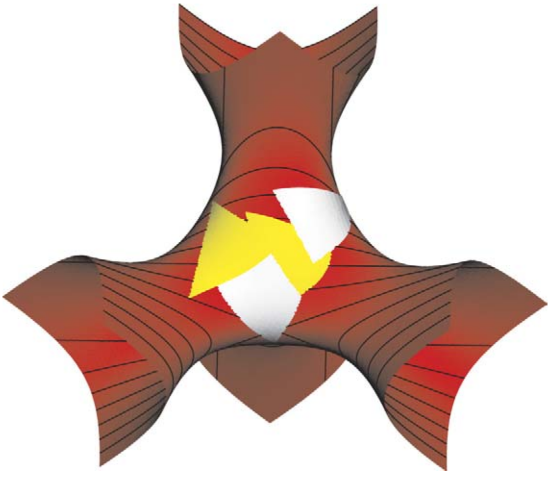

Fig. 4. Implicit surface with parametrized patches.

Table 1
Numerical experiment using the two-curve technique

Error level	Distance of P, P_0	$F(P_g)$
10^{-10}	10^{-9}	10^{-8}
10^{-15}	10^{-14}	10^{-13}
10^{-20}	10^{-18}	10^{-18}
10^{-30}	10^{-28}	10^{-28}
10^{-40}	10^{-37}	10^{-37}
10^{-50}	10^{-47}	10^{-48}

Similarly, $Q_2^p(0.77339 : 0.19981 : -0.39962)$ is a point on K_2, which corresponds to the point

$$Q_2(0.23467 : -0.021736 : 0.32529 : 0.41785)$$

on S. The parametrization of C_2 is

$$\begin{pmatrix} -1.05829t_2^3 + 3.27466t_2^2 + 0.80973t_2 - 5.21309 \\ -0.85279t_2^3 - 0.88123t_2^2 + 4.94149t_2 - 2.14387 \\ 3.18407t_2^2 - 9.35177t_2 + 7.01474 \\ 5.18538t_2^2 - 10.67940t_2 + 4.04335 \end{pmatrix}.$$

Let the coordinates of the curve C_1 be $(C_{1,x_1}, C_{1,x_2}, C_{1,x_3}, C_{1,x_4})$ and the coordinates of C_2 be $(C_{2,x_1}, C_{2,x_2}, C_{2,x_3}, C_{2,x_4})$. Let the equation of the line connecting $C_1(t_1)$ and $C_2(t_2)$ be $C_{1,x_i} + \lambda C_{2,x_i}$. Substituting this equation into F we get $B_1(t_1, t_2)\lambda + B_2(t_1, t_2)\lambda^2$. Thus $\lambda = -B_1/B_2$. Substituting it back into $C_{1,x_i}(t_1) + \lambda(t_1, t_2)C_{2,x_i}(t_2)$ gives a parametrization of a neighborhood of the point P.

Fig. 4 shows several parametrized patches on a given cubic surface.

Table 1 shows the numerical behavior of the two-curve parametrization technique in the case of Example 1. The letters P, P_0, P_g denote the given point on the surface S, the point generated by the parametrization for $(0, 0)$ parameter values, and a point generated by the parametrization for $(2, -5)$ parameter values. We see if the error goes to zero, then P_0 converges to P and $F(P_g)$ converges to zero.

Now we prove the correctness of Algorithm 2.

Theorem 10. *For a nonsingular cubic surface S and generic point $P \in S$, Algorithm 2 produces a regular local parametrization if and only if P is a point with the t-property.*

Proof. If P has the t-property, then the whole component containing P has only points with this property. Thus we can always find lines through P which intersect the surface S in two additional real intersections with the non-convex component of the surface, i.e., in points with the t-property.

Due to the construction of the algorithm, the image of the origin is P and the image of the map is contained in the surface.

It remains to be shown that – in step 3 – it is always possible to choose a point Q_2 on the contour curve such that the tangent lines to the curves C_{Q_1} and C_{Q_2} are not coplanar.

Let l_{P_1} denote the tangent line at P_1 to the curve C_{Q_1}. (l_{P_1} is the intersection of the tangent planes $T_{Q_1}S$ and $T_{P_1}S$.) Furthermore denote by $l^j_{P_2}$ the tangent line at P_2 to the curve $C_{Q^j_2}$. (It is the intersection of the tangent planes $T_{Q^j_2}S$ and $T_{P_2}S$.) The line connecting P_2 with the intersection of l_{P_1} and $T_{P_2}S$ gives the tangent direction at P_2 which is forbidden. We have to show that it is always possible to choose a point Q^j_2 on the contour with respect to P_2 such that $l^j_{P_2}$ is not the forbidden direction.

We show that it is not possible to have the same tangent direction for all points on the apparent contour with respect to P_2. For each point on the apparent contour we get a corresponding point on the contour generator Q^j_2 and a tangent direction $l^j_{P_2}$. If all points gave the same tangent direction l_{P_2}, then all tangent planes $T_{Q^j_2}S$ would go through this line, i.e. we would get a pencil of planes. Thus the envelope surface of these tangent planes would degenerate into a line, which is not possible.

As one can verify by direct computation, if the tangents to the curves C_{Q_1} and C_{Q_2} are not coplanar, then the Jacobian of the parametrization has full rank at P.

On the other hand, if P does not have the t-property, then Algorithm 2 stops in step 1. In this situation it is clear that any line through P would intersect S in another point which does not have the t-property. □

Remark 1. It can be shown that the parametrization computed by Algorithm 2 has bidegree $(6, 6)$ and total degree 12.

We call a number $k \in \mathbb{N}$ the *index* of the parametrization if all points outside a Zariski closed subset are generated by k complex parameter pairs (Sendra and Winkler, 2001). A proper parametrization has index 1.

Proposition 11. *The index of the parametrization obtained by the two-curve technique equals 6.*

Idea of the proof. If P is a point on S that can be parametrized using the points Q_1, Q_2, we have to compute how many lines through P exist, which intersects both curves C_{Q_1}, C_{Q_2}. As the planes of the two curves C_{Q_1}, C_{Q_2} intersect in a line, the curves have three intersections on a line. If we project the two curves C_{Q_1}, C_{Q_2} from P on an arbitrary plane we get nine intersection points from which three are on a line (Bezout's theorem). Hence we can reach P six times using this parametrization method. See Manin (1986) for a complete proof. □

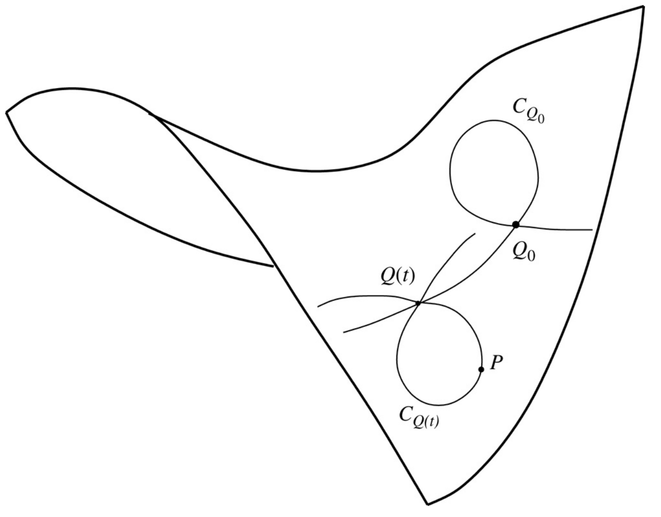

Fig. 5. The repeater technique.

Remark 2. Methods for reducing the index of the parametrization of curves exist (Sederberg, 1984, 1986). Unfortunately, currently no methods for reducing the index in the surface case are available.

4.3. The repeater technique

Here is an alternative idea for computing a local parametrization. We do not describe it in full detail because it is inferior to the two-curve technique in two ways. First, it is applicable in less situations, second it is more complicated to analyze the degeneracy conditions expressing the vanishing of the Jacobian.

Let Q_0 be a real point on S as in Fig. 5. The rational cubic $C_{Q_0} = T_{Q_0}S \cap S$ has a rational parametrization $\pi_{Q_0} : \mathbb{R} \to C_{Q_0}$. Let $C_{Q(t)}$ be a curve cut by the tangent plane at the point $Q(t) := \pi_{Q_0}(t)$. Then, the parametrization of the curve $C_{Q(t)}$, $\pi_{Q(t)} : \mathbb{R} \to C_{Q(t)}, s \mapsto \pi_{Q(t)}(s)$ gives a parametrization of a neighborhood of the point $P := \pi_{Q(t)}(s)$.

The above technique leads to the following algorithm.

Algorithm 3 (``repeater technique'')
Given: An implicit equation F of a cubic surface S and a point $P \in S$.
Synopsis: Find four polynomials depending on two parameters, which give a local parametrization of S around P.

(1) Check the t-property for P
  (a) If P does not have the t-property the algorithm stops.
  (b) If P has the t-property go to the next step.
(2) Compute the contour generator K with respect to P and choose a point Q on it with the t-property (see Algorithm 1).
  (a) If there is no such point the algorithm stops.
  (b) If there is such a point go to the next step.
(3) Compute the contour generator with respect to Q and choose a point Q_0 on it such that $T_{Q_0}S$ does not contain the tangent at Q to K.

(4) Compute the intersection of S with tangent plane $T_{Q_0}S$: C_{Q_0}.
(5) Parametrize C_{Q_0}, such that $Q = \pi_{Q_0}(0)$.
(6) Parametrize the curve $C_{Q(t)}$, which is the intersection of S with the tangent plane at the point $Q(t) := \pi_{Q_0}(t)$, such that $P = \pi_{Q(0)}(0)$.

Remark 3. It may happen that the contour generator with respect to the point P lies on the convex component of S. In this case the algorithm stops at step 2. In other words the t-property for P is not sufficient for Algorithm 3 to produce a result.

Remark 4. It is not difficult to prove that Algorithm 3 always produces a local parametrization, if the contour generator with respect to P has points with the t-property. In order to obtain a *regular* local parametrization we need an additional condition analogous to the condition in Algorithm 2 that the tangents at P_1, P_2 to C_{Q_1} and C_{Q_2} are not coplanar. More precisely, the tangent plane at $T_{Q_0}S$ must not contain the tangent at Q to the contour generator with respect to P. We do not give a detailed proof, as the repeater technique is less useful than the two-curve technique for our purposes.

Remark 5. The total degree of the parametrization using Algorithm 3 is 12. The index of the parametrization obtained by the repeater technique is 6, see Manin (1986).

4.4. The reflection technique

Algorithms 2 and 3 fail if the surface has two components, and the point around which we want to compute a local parametrization is on the convex piece. We can detect this case simply by checking if the point has the t-property or not. If it does not have this property, then it is located on the convex component. In such situations, we can use the following technique.

Let P be the point on the surface S; see Fig. 6. Using Algorithm 2 we can parametrize some region of S. Connect the point P with any point Q from the parametrized region and denote by C the further intersection point with S. From C reflect the points of the parametrized region. This gives a parametrization of the neighborhood of the point P.

We summarize this idea in

Algorithm 4 (``reflection technique'')
Given: An implicit equation F of a cubic surface S and a point $P \in S$.
Synopsis: Find four polynomials depending on two parameters, which define a parametrization of a neighborhood of P.

(1) Using Algorithm 2 compute a local parametrization for a point $Q \in S$ with the t-property, where $Q \notin T_P S$. Let the parametrization be $P_{t_1,t_2} := (X(t_1,t_2) : Y(t_1,t_2) : Z(t_1,t_2) : W(t_1,t_2))$
(2) Connect P with the point Q, and intersect this line with S. Let $C(c_1 : c_2 : c_3 : c_4)$ be the further intersection point.
(3) Intersect the line CP_{t_1,t_2} with S. Compute the third point of intersection in terms of t_1, t_2.

Remark 6. In step 3 in Algorithm 4 we need to compute the third point of intersection of S with a line through two points of S. This can be done in the same way as in step 6 of Algorithm 2.

Example 2. We use the same surface as in Example 1. We want to construct a local parametrization around $P = (1.53295 : 53.20912 : 10.85109 : 1)$ Using Algorithm 2 we compute a local parametrization around the point $Q = (1 : 3 : 27 : 27)$ as in Example 1. Let us

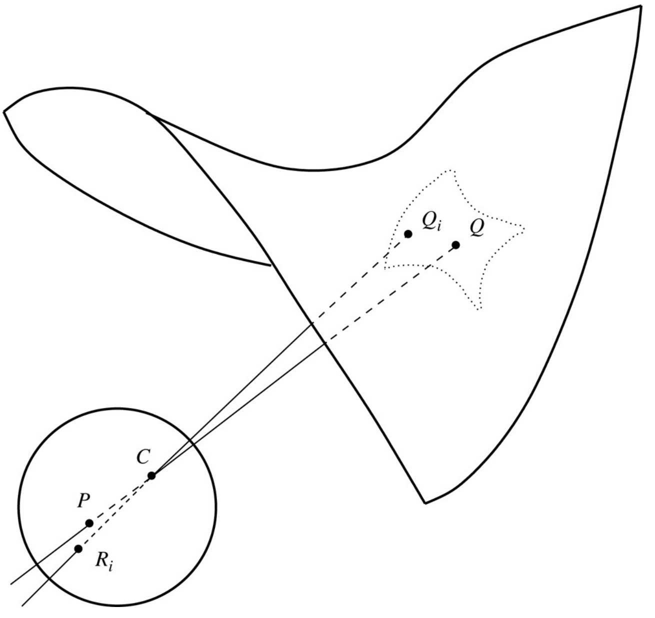

Fig. 6. The reflection technique.

denote the computed parametrization by $(X(t_1, t_2) : Y(t_1, t_2) : Z(t_1, t_2) : W(t_1, t_2))$. The parametric equation of the line connecting P and Q is

$$(u + 1.53295 : 3u + 53.20912 : 27u + 10.85109 : 27u + 1).$$

Substituting it into the equation of S, and solving the resulting equation for u we get the further intersection of the line with S:

$$C(-2.46836 : 41.20518 : -97.18435 : -107.03544).$$

The line l_{CQ_i} connecting C with the points of the parametrized region has the form

$$(\lambda X - 2.46836 : \lambda Y + 41.20518 : \lambda Z - 97.18435 : \lambda W - 107.03544).$$

Substituting it into F we get an equation of the form $B_1(t_1, t_2)\lambda + B_2(t_1, t_2)\lambda^2$. Computing λ and substituting it back into the equation of l_{CQ_i}, we get a local parametrization around P.

Table 2 shows the numerical behavior of the reflection technique in the case of Example 2. The letters P, P_0, P_g again stand for the given point on the surface S, the point generated by the parametrization for $(0, 0)$ parameter values, and a point corresponding to the $(2, -5)$ parameter values. We can observe if the error goes to zero, then P_0 converges to P and $F(P_g)$ converges to zero.

Theorem 12. *For a nonsingular cubic surface S and point $P \in S$, Algorithm 4 always gives a regular local parametrization.*

Proof. We have to show that it is always possible to find $R \in S$ with the t-property, where $R \notin T_P S$. The intersection of S and $T_P S$ is a degree 3 curve. The non-convex component of S contains other points, and these points have the t-property. Let R be one of these points. As it was shown before, Algorithm 2 produces a regular local parametrization around the point R.

Table 2
Numerical experiment using the reflection technique

Error level	Distance of P, P_0	$F(P_g)$
10^{-10}	10^{-8}	10^{-7}
10^{-15}	10^{-13}	10^{-12}
10^{-20}	10^{-18}	10^{-17}
10^{-30}	10^{-27}	10^{-27}
10^{-40}	10^{-37}	10^{-37}
10^{-50}	10^{-47}	10^{-47}

As $R \notin T_P S$, the line connecting P and R is not tangent at P. Let C be the third point of intersection of S with the line PR. Then the line PR intersects the two tangent planes at P and R transversally. This implies that the reflection of the cubic surface S at C restricts to a local isomorphism of sufficiently small regions of S around P and R. Therefore the composition of this map with the regular local parametrization around R is regular. □

The reflection technique works for any type of cubic surfaces. It can be applied arbitrarily, but it is particularly interesting in the situation, when the given surface has two components, and the point around which we want to compute a local parametrization lies on the convex part.

Remark 7. As one may verify by a straightforward computation, the parametrization computed by Algorithm 4 has bidegree (12, 12) and total degree 24.

Proposition 13. *The index of the parametrization obtained by the reflection technique is 6.*

Proof. Reflection at a point is birational and does not change the index. As the index of the two-curve technique is 6, the index of the reflection technique is also 6. □

4.5. Covering a surface by local parametrizations

Theorem 14. *Given a nonsingular cubic surface. It can be covered by a finite number of local parametrizations.*

Proof. For each point P on the surface we can compute a local parametrization $\mathcal{P}_P$, which covers some open neighborhood U_P of P. Obviously $S = \cup_{P \in S} U_P$. Since S is compact, there exist a finite subcover. □

4.6. Continuity failures

The presented algorithms branch due to the sign of the value of an arithmetic expression. Instead of exact values only approximations are computed. Therefore the sign of a real valued expression might be evaluated incorrectly. This might lead to incorrect decision in the diverging step. Incorrect decisions finally result in incorrect outputs. Showing continuity is straightforward under the assumptions that no wrong decisions are made. But in the presence of wrong decisions continuity is rather an exception than a rule.

Most of the operations of the presented algorithms can be made continuous, but there are some operations where we ran into troubles. We demonstrate the difficulties with an example: choose a point on the apparent contour, i.e. determine a point on a planar degree four nonsingular curve.

Our idea to solve this problem was the following: compute the discriminant of the curve and the real zeros of the discriminant. Using the computed real roots we determine intervals where

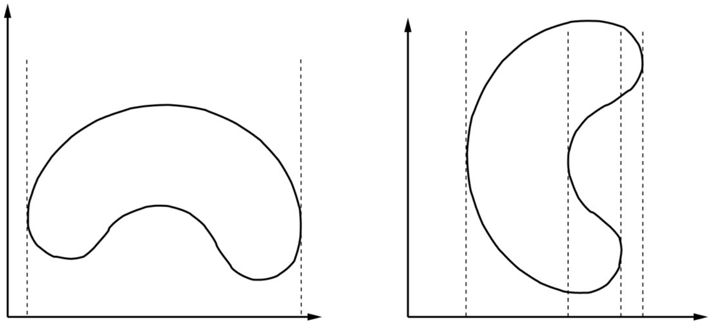

Fig. 7. An example of a quartic curve for which the number of real roots of the discriminant changes for continuous changes of the coordinates.

the discriminant has negative sign. Substituting the middle point of the first interval into the equation of the curve, and taking the smallest solution of the arising univariate polynomial gives a point on the curve. However, this algorithm is not continuous on the set of coefficients of planar degree four nonsingular curves.

In fact one can construct an example of a nonsingular quartic (see Fig. 7) for which the number of real roots of the discriminant changes for continuous changes of the coordinates.

5. Conclusion

Using the techniques described in this paper, the vicinity of a generic point on a given surface can be covered by a regular rational parametrization. As a potential advantage, this parametrization is found without analyzing the type of the cubic surface, i.e., without discussing the system of the 27 lines.

It can be expected that the results extend to non-generic regular points and to singular surfaces. However, the complete analysis of the singular cases is beyond the scope of this paper, since it requires the study of each surface class separately (20 cases over $\mathbb{C}$, and many more cases over $\mathbb{R}$). Further results will be presented in Szilágyi (2005).

As a matter of future research, the numerical stability of the method should be further explored. More precisely, the following two questions should be addressed: Firstly, since it is generally not theoretically justified to use this methods with floating point numbers, it would be interesting to develop criteria which help to decide whether the algorithm can be applied to specific data or not. More precisely, it would be interesting to get additional, possibly data-dependent, conditions, which guarantee that the algorithm works reasonably well for certain specific sets of input data. We expect that results in this direction could be achieved, e.g., using techniques from interval arithmetics.

Secondly, the quality of the resulting local parametrizations should be analyzed. As observed in our experiments, this quality can greatly be enhanced. by using some heuristic ideas for optimizing the position of the randomly chosen lines, etc. In addition, whenever a special coordinate system has to be chosen, we use an orthogonal transformation of the homogeneous coordinates, in order to minimize the effect of rounding errors.

Another challenging problem is the use of exact symbolic computation throughout the algorithm. While the current implementation had to resort to floating point numbers, the underlying concepts are purely symbolic and would therefore benefit from a symbolic-computation-based implementation.

Finally, since the method produces improper parametrizations of a relatively high index, systematic techniques for reducing the index of a parametrization could be of some interest and should be explored.

Acknowledgments

The authors are indebted to the anonymous referees, and to R. Sendra for their fruitful remarks and comments, which have greatly helped to improve the manuscript. This research has been supported by the Austrian science fund (FWF) in the frame of the special research area SFB 013 on “Numerical and Symbolic Scientific Computing”.

References

Abhyankar, S.S., Bajaj, C.L., 1988. Automatic parameterization of rational curves and surfaces. III. Algebraic plane curves. Comput. Aided Geom. Design 5 (4), 309–321.
Abhyanker, S.S., Bajaj, C., 1987. Automatic parametrization of rational curves and surfaces. II. Cubics and cubicoids. Comput. Aided Des. 19 (9), 499–502.
Bajaj, C.L., Holt, R.J., Netravali, A.R., 1998. Rational parametrization of non-singular real cubic surfaces. ACM Trans. Graph. 17, 1–31.
Berry, T.G., Patterson, R.R., 2001. Implicitization and parametrization of nonsingular cubic surfaces. Comput. Aided Geom. Design 18 (8), 723–738.
Bruce, J.W., Wall, C.T.C., 1979. On the classification of cubic surfaces. J. London Math. Soc. (2) 19 (2), 245–256.
Buchberger, B., 1988. Applications of Gröbner bases in nonlinear computational geometry. In: Trends in Computer Algebra. Springer, Berlin, pp. 52–80.
Busé, L., 2001. Residual resultant over the projective plane and the implicitization problem. In: Proc. ISSAC. ACM, New York, pp. 48–55.
Cipolla, R., Giblin, P., 2000. Visual Motion of Curves and Surfaces. Cambridge University Press, Cambridge.
Corless, R.M., Giesbrecht, M.W., Kotsireas, I.S., Watt, S.M., 2001. Numerical implicitization of parametric hypersurfaces with linear algebra. In: AISC 2000. In: LNCS, Springer, Berlin, pp. 174–183.
Coste, M., Roy, M.-F., 1988. Thom’s lemma, the coding of real algebraic numbers and the computation of the topology of semi-algebraic sets. J. Symbolic Comput. 5 (1–2), 121–129.
Dokken, T., 2001. Approximate implicitization. In: Mathematical Methods for Curves and Surfaces. Vanderbilt Univ. Press, Nashville, TN, pp. 81–102.
Dokken, T. et al., 2001. Intersection algorithms for geometry-based it applications using approximate algebraic methods. (GAIA II), EU project no. IST-2001-35512.
Gonzalez-Vega, L., Necula, I., 2002. Efficient topology determination of implicitly defined algebraic plane curves. Comput. Aided Geom. Design 19 (9), 719–743.
Henderson, A., 1960. The Twenty-seven Lines upon the Cubic Surface. Hafner, New York.
Hoschek, J., Lasser, D., 1993. Fundamentals of Computer Aided Geometric Design. AK Peters, Wellesley, MA.
Kendig, K., 1977. Elementary Algebraic Geometry. Springer, New York.
Manin, Y.I., 1986. Cubic Forms. North-Holland, Amsterdam.
Pérez-Díaz, S., Sendra, J., Sendra, J.R., 2004. Parametrization of approximate algebraic curves by lines. Theoret. Comput. Sci. 315 (2–3), 627–650.
Pérez-Díaz, S., Sendra, J., Sendra, J.R., 2005. Parametrization of approximate algebraic surfaces by lines. Comput. Aided Geom. Design 22 (2), 147–181.
Schicho, J., 1998a. Rational parameterization of real algebraic surfaces. In: Proc. ISSAC. ACM, New York, pp. 302–308.
Schicho, J., 1998b. Rational parametrization of surfaces. J. Symbolic Comput. 26 (1), 1–29.
Schirra, S., 2000. Robustness and precision issues in geometric computation. In: Handbook of Computational Geometry. North-Holland, Amsterdam, pp. 597–632.
Sederberg, T.W., 1984. Degenerate parametric curves. Comput. Aided Geom. Design 1 (4), 301–307.
Sederberg, T.W., 1986. Improperly parameterized rational curves. Comput. Aided Geom. Design 3 (1), 67–75.
Sederberg, T.W., 1990. Techniques for cubic algebraic surfaces. II. IEEE Comput. Graph. Appl. 10 (5), 12–21.
Sederberg, T.W., Snively, J.P., 1987. Parametrization of cubic algebraic surfaces. In: The Mathematics of Surfaces, II. Oxford Univ. Press, pp. 299–319.

Segre, B., 1942. The Non-singular Cubic Surfaces. Oxford University Press.

Sendra, J.R., Winkler, F., 2001. Tracing index of rational curve parametrizations. Comput. Aided Geom. Design 18 (8), 771–795.

Szilágyi, I., 2005. Symbolic–numeric techniques for cubic surfaces. Ph.D. Thesis (submitted).

Zheng, J., Sederberg, T.W., Chionh, E.-W., Cox, D.A., 2003. Implicitizing rational surfaces with base points using the method of moving surfaces. In: Topics in Algebraic Geometry and Geometric Modeling. AMS, Providence, RI, pp. 151–168.